WHAT IS CLAIMED IS:

1.   A system, comprising:

5   a processor; and

a memory comprising program instructions, wherein the program instructions are executable by the processor to implement a program configured to:

10   call an API function of a library; and

obtain an error trace for the API function;

wherein the error trace includes one or more error trace elements, wherein each
15   error trace element includes information describing a particular error generated during execution of the API function.

2.   The system as recited in claim 1, wherein the API function is configured to call a plurality of library functions in a function call stack, wherein each of the plurality of
20   library functions is configured to, if the library function generates an error, add an error trace element to the error trace.

3.   The system as recited in claim 1, wherein the API function is configured to call another library function, wherein the other library function is configured to, if the
25   other library function generates an error, add an error trace element to the error trace.

4.   The system as recited in claim 1, wherein the error trace is a program structure defined in the library, and wherein the error trace further includes a field indicating a count of the error trace elements in the error trace.

30

5.   The system as recited in claim 1, wherein each error trace element indicates one or more of a location where the particular error of the error trace element occurred, an error type of the particular error, and what the particular error is.

6.   The system as recited in claim 5, wherein the location of the particular error includes one or more of a function name, a source file name, and a line number where the particular error occurred.

7.   The system as recited in claim 1, wherein the program is further configured to determine from the error trace element one or more of a location where the particular error of the error trace element occurred, an error type of the particular error, and what the particular error is.

8.   The system as recited in claim 1, wherein the library is a C/C++ interface library.

9.   A system, comprising:

a processor; and

a memory comprising program instructions, wherein the program instructions are executable by the processor to implement a library and a program configured to call a library function of the library;

wherein the library function is configured to, if the library function generates one or more errors, add an error trace element to an error trace for each error, wherein each error trace element includes information describing a particular error generated during execution of the library function; and

wherein, after completion of the library function, the program is further

configured to obtain the error trace for the library function..

10.   The system as recited in claim 9, wherein the library function is configured to call a plurality of library functions in a function call stack, wherein each of the plurality of library functions is configured to, if the library function generates an error, add an error trace element to the error trace.

11.   The system as recited in claim 9, wherein the program is further configured to determine from the error trace element one or more of a location where each error occurred, an error type of each error, and what each error is.

12.   The system as recited in claim 9, wherein the library is a C/C++ interface library.

13.   A system, comprising:

a processor; and

a memory comprising program instructions, wherein the program instructions are executable by the processor to implement a program configured to:

call a function; and

obtain an error trace for the function;

wherein the error trace includes one or more error trace elements, wherein each error trace element includes information describing a particular error generated during execution of the function.

14.   The system as recited in claim 13, wherein the function is configured to

call a plurality of functions in a function call stack, wherein each of the plurality of functions is configured to, if the particular function generates an error, add an error trace element to the error trace.

15.   The system as recited in claim 13, wherein the function is configured to call another function, wherein the other function is configured to, if the other function generates an error, add an error trace element to the error trace.

16.   The system as recited in claim 13, wherein the error trace further includes a field indicating a count of the error trace elements in the error trace.

17.   The system as recited in claim 13, wherein each error trace element indicates one or more of a location where the particular error of the error trace element occurred, an error type of the particular error, and what the particular error is.

18.   The system as recited in claim 17, wherein the location of the particular error includes one or more of a function name, a source file name, and a line number where the particular error occurred.

19.   The system as recited in claim 13, wherein the program is further configured to determine from the error trace element one or more of a location where the particular error of the error trace element occurred, an error type of the particular error, and what the particular error is.

20.   A system, comprising:

a processor; and

a memory comprising program instructions, wherein the program instructions are executable by the processor to implement a program configured to:

call a function;

determine if the function generated an error; and

if the function generated an error, obtain an error trace for the function, wherein the error trace includes one or more error trace elements, wherein each error trace element includes information describing a particular error generated during execution of the function.

21.   The system as recited in claim 20, wherein the function is configured to call a plurality of functions in a function call stack, wherein each of the plurality of functions is configured to, if the particular function generates an error, add an error trace element to the error trace.

22.   The system as recited in claim 20, wherein each error trace element indicates one or more of a location where the particular error of the error trace element occurred, an error type of the particular error, and what the particular error is.

23.   The system as recited in claim 20, wherein the program is further configured to determine from the error trace element one or more of a location where the particular error of the error trace element occurred, an error type of the particular error, and what the particular error is.

24.   The system as recited in claim 20, wherein the function is a function of a library, and wherein said call is to an API call of the function in an API to the library.

25.   The system as recited in claim 24, wherein the library is a C/C++ interface library.

26.   A system, comprising:

a processor; and

5   a memory comprising program instructions, wherein the program instructions are
        executable by the processor to implement a program configured to
        generate code configured to:

call an API function of a library; and

10

obtain an error trace for the API function;

wherein the error trace includes one or more error trace elements, wherein each
        error trace element includes information describing a particular error
15   generated during execution of the API function.

27.   The system as recited in claim 26, wherein the API function is configured
to call a plurality of library functions in a function call stack, wherein each of the plurality
of library functions is configured to, if the library function generates an error, add an error
20   trace element to the error trace.

28.   The system as recited in claim 26, wherein each error trace element
indicates one or more of a location where the particular error of the error trace element
occurred, an error type of the particular error, and what the particular error is.

25

29.   The system as recited in claim 26, wherein the code is further configured
to determine from the error trace element one or more of a location where the particular
error of the error trace element occurred, an error type of the particular error, and what the
particular error is.

30

30. The system as recited in claim 26, wherein the library is a C/C++ interface library.

31. A system, comprising:

5

a processor; and

a memory comprising program instructions, wherein the program instructions are executable by the processor to implement a library comprising one or more

10 library functions and an API to the library, wherein the API includes:

one or more function definitions configured for access of the one or more library functions by a program; and

15 a function definition for a get error trace function configured for access by the program to get error traces generated by the one or more library functions;

wherein each error trace includes one or more error trace elements, wherein each

20 error trace element includes information describing a particular error generated during execution of the associated library function.

32. The system as recited in claim 31, wherein one of the library functions is configured to call a plurality of other library functions in a function call stack, wherein

25 each of the plurality of library functions is configured to, if the library function generates an error, add an error trace element to the error trace.

33. The system as recited in claim 32, wherein the location of the particular error includes one or more of a function name, a source file name, and a line number

30 where the particular error occurred.

34.   The system as recited in claim 31, wherein the program is further configured to determine from the error trace element one or more of a location where the particular error of the error trace element occurred, an error type of the particular error, and what the particular error is.

5

35.   The system as recited in claim 31, wherein the library is a C/C++ interface library.

10   36.   A system, comprising:

means for a plurality of functions in a function call stack to generate information describing one or more errors generated by the plurality of library functions;

15

means to obtain the generated information; and

means to determine from the obtained information one or more of a location where each error occurred, an error type of each error, and what the each error is.

20

37.   The system as recited in claim 36, wherein the plurality of functions are functions of a library, further comprising means to call the plurality of functions in the function call stack from a program.

25

38.   The system as recited in claim 37, wherein the library is a C/C++ interface library.

39.   A method, comprising:

30

a program calling a function;

the program determining if the function generated an error; and

5   if the function generated an error, the program obtaining an error trace for the
          function;

wherein the error trace includes one or more error trace elements, wherein each
          error trace element includes information describing a particular error
10   generated during execution of the function.

40.   The method as recited in claim 39, further comprising:

the function calling a plurality of functions in a function call stack;

15

for each of the plurality of functions, if the particular function generates an error,
          adding an error trace element to the error trace.

41.   The method as recited in claim 39, further comprising:

20

the function calling another function;

if the other function generates an error, adding an error trace element to the error
          trace.

25

42.   The method as recited in claim 39, wherein each error trace element
indicates one or more of a location where the particular error of the error trace element
occurred, an error type of the particular error, and what the particular error is.

30   43.   The method as recited in claim 42, wherein the location of the particular

error includes one or more of a function name, a source file name, and a line number where the particular error occurred.

44.   The method as recited in claim 39, further comprising determining from the error trace element one or more of a location where the particular error of the error trace element occurred, an error type of the particular error, and what the particular error is.

45.   The method as recited in claim 39, wherein the function is a function of a library called via an API to the library.

46.   The method as recited in claim 45, wherein the library is a C/C++ interface library.

47.   A method, comprising:

a program calling a library function of a library via an API to the library;

if the library function generates one or more errors, adding an error trace element to an error trace for each error;

after completion of the library function, the program obtaining the error trace for the library function;

wherein each error trace element includes information describing a particular error generated during execution of the library function.

48.   The method as recited in claim 47, further comprising:

the library function calling a plurality of library functions in a function call stack;

for each of the plurality of library functions, if the library function generates an error, adding an error trace element to the error trace.

49.   The method as recited in claim 47, further comprising determining from the error trace element one or more of a location where each error occurred, an error type of each error, and what each error is.

50.   The method as recited in claim 47, wherein the library is a C/C++ interface library.

51.   A computer-accessible medium comprising program instructions, wherein the program instructions are configured to implement:

a program calling a function;

the program determining if the function generated an error; and

if the function generated an error, the program obtaining an error trace for the function;

wherein the error trace includes one or more error trace elements, wherein each error trace element includes information describing a particular error generated during execution of the function.

52.   The computer-accessible medium as recited in claim 51, wherein the program instructions are further configured to implement:

the function calling a plurality of functions in a function call stack;

for each of the plurality of functions, if the particular function generates an error, adding an error trace element to the error trace.

53.   The computer-accessible medium as recited in claim 51, wherein the program instructions are further configured to implement:

the function calling another function;

if the other function generates an error, adding an error trace element to the error trace.

54.   The computer-accessible medium as recited in claim 51, wherein each error trace element indicates one or more of a location where the particular error of the error trace element occurred, an error type of the particular error, and what the particular error is.

55.   The computer-accessible medium as recited in claim 54, wherein the location of the particular error includes one or more of a function name, a source file name, and a line number where the particular error occurred.

56.   The computer-accessible medium as recited in claim 51, wherein the program instructions are further configured to implement determining from the error trace element one or more of a location where the particular error of the error trace element occurred, an error type of the particular error, and what the particular error is.

57.   The computer-accessible medium as recited in claim 51, wherein the function is a function of a library called via an API to the library.

58.   The computer-accessible medium as recited in claim 57, wherein the library is a C/C++ interface library.

59.   A computer-accessible medium comprising program instructions, wherein the program instructions are configured to implement:

5   a program calling a library function of a library via an API to the library;

      if the library function generates one or more errors, adding an error trace element to an error trace for each error;

10   after completion of the library function, the program obtaining the error trace for the library function;

      wherein each error trace element includes information describing a particular error generated during execution of the library function.

15

60.   The computer-accessible medium as recited in claim 59, wherein the program instructions are further configured to implement:

      the library function calling a plurality of library functions in a function call stack;

20

      for each of the plurality of library functions, if the library function generates an error, adding an error trace element to the error trace.

61.   The computer-accessible medium as recited in claim 59, wherein the
25   program instructions are further configured to implement determining from the error trace element one or more of a location where each error occurred, an error type of each error, and what each error is.

62.   The computer-accessible medium as recited in claim 59, wherein the
30   library is a C/C++ interface library.